PPD/EED

Cold Dark Matter Survey (CDMS)

CDMS System Test Module (CDMS-STM)

Version 1

Preliminary Technical Specification

by
Mark Kozlovsky

Based upon a draft specification written
by Mike Crisler on 08/17/01

January 30, 2002

Batavia, Illinois

1. Introduction

The main purpose of the CDMS/STM is to simulate CDMS detector electronics in order to verify the entire data handling system including ZIP modules, ZIP RTF modules, and the DAQ subsystem.

The CDMS/STM is going to replace all existing Mini-Bob units used to test ZIP modules and the Auxiliary Boards, which serve as providers of Variable +/- signals on crate backplanes.

To achieve its goals the CDMS/STM will generate simulated charge and phonon detector pulses with programmable amplitudes and rising times, and redirect specified I/O signals from tested ZIP modules to four test BNC connectors on its front panel.

The CDMS/STM control circuitry will be built around a 16-bit microcontroller running embedded software.

2. Module Specification

- **Modes of Operation:**

a) **Automated Test (Auto)** – an internally synthesized sine wave is used to test ZIP modules in accordance with a pre-programmed test configuration.
b) **Free Running Test (Run)** – a pre-programmed Event Sequence is used to test the system. Event rate is programmed in the Event Sequence and may change from one event to another. If no rate is programmed for a given event, then it is assumed that this event is running at 10 Hz rate. The test is running continuously going repeatedly through the Event Sequence. It starts and stops by commands from the system. When being terminated, the test stops only after completing the Event Sequence.
c) **Synchronized Test (Synch)** – a pre-programmed Event Sequence is used to test the system. Events are synchronized with external trigger signals. The test proceeds synchronously with those signals going through preprogrammed events in the Event Sequence as in Run mode but ignores the programmed rate for each event. It wraps around when number of triggers exceeds the length of the Event Sequence. The incoming trigger may be gated in this mode by external commands, thus making possible to run the test from any continuously running pulse source.
d) **Single Step Test (Single)** – any one event from the Event Sequence is used to test the system. The test starts by an external command. When the start command is issued continuously, all events in this mode are the same.
e) Continuous Single Step Test (Cont) – same as Single mode but synchronized with external trigger signals, which may be gated by external commands.

Comments: all above modes can be set remotely via the 96-pin rear connector or locally via mode switches on the module's front panel. Remote settings always take precedent over the local ones.

- **Tested Signals:**

Tested signals are those ZIP output and input signals that are redirected to the BNC connectors on the CDMS/STM module's front panel for further analysis with standard measurement instruments, such as an oscilloscope, etc.

Tested signals are organized in four groups named in accordance with their BNC output connections:

Test Point 1 (TP-1):

1.	Squid A	- Output of Squid A Driver	- P3/3
2.	Phonon A Offset	- Driver A Offset	- P3/9
3.	Qamp Outer	- Charge Amplifier Driver Output	- P3/1
4.	Phonon Pulser A	- Internal Phonon Pulser A	- none
5.	Squid A Bias	- Squid A Bias Input	- P1/4
6.	QET Bias A	- Squid A QET Input	- P1/3
7.	Qbias Outer	- Charge Amplifier Bias Input	- P1/11
8.	Sine Wave	- Sine Wave Generator Output	- none

Test Point 2 (TP-2):

1.	Squid B	- Output of Squid B Driver	- P3/4
2.	Phonon B Offset	- Driver B Offset	- P3/10
3.	Qamp Inner	- Charge Amplifier Driver Output	- P3/2
4.	Phonon Pulser B	- Internal Phonon Pulser B	- none
5.	Squid B Bias	- Squid B Bias Input	- P1/19
6.	QET Bias B	- Squid B QET Input	- P1/35
7.	Qbias Outer	- Charge Amplifier Bias Input	- P1/15
8.	Not connected		- none

Test Point 3 (TP-3):

1.	Squid C	- Output of Squid C Driver	- P3/5
2.	Phonon C Offset	- Driver C Offset	- P3/11
3.	Qamp Outer Pulser	- Internal Charge Pulser Output	- none
4.	Phonon Pulser C	- Internal Phonon Pulser C	- none
5.	Squid C Bias	- Squid C Bias Input	- P1/8
6.	QET Bias C	- Squid C QET Input	- P1/7
7.	LED 1	- LED Driver Output 1	- P1/42
8.	FET Heater	- Target Heater	- P1/45

Test Point 4 (TP-4):

1. Squid D	- Output of Squid D Driver	- P3/6
2. Phonon D Offset	- Driver D Offset	- P3/12
3. Qamp Inner Pulser	- Internal Charge Pulser Input	- none
4. Phonon Pulser D	- Internal Phonon Pulser D	- none
5. Squid D Bias	- Squid D Bias Input	- P1/23
6. QET Bias D	- Squid D QET Input	- P1/38
7. LED2	- LED Driver Output 2	- P1/26/25
8. Not connected		- none

Each tested signal has a corresponding ground reference connected to a ground pin of a BNC connector for proper measurements. This is shown bellow:

Test Point 1 (TP-1):

1. Squid A	- AGND
2. Phonon A Offset	- AGND
3. Qamp Outer	- AGND
4. Phonon Pulser A	- AGND
5. Squid A Bias	- FEGND
6. QET Bias A	- FEGND
7. Qbias Outer	- FEGND
8. Sine Wave	- FEGND

Test Point 2 (TP-2):

1. Squid B	- AGND
2. Phonon B Offset	- AGND
3. Qamp Inner	- AGND
4. Phonon Pulser B	- AGND
5. Squid B Bias	- FEGND
6. QET Bias B	- FEGND
7. Qbias Outer	- FEGND
8. None	- FEGND

Test Point 3 (TP-3):

1. Squid C	- AGND
2. Phonon C Offset	- AGND
3. Qamp Outer Pulser	- AGND

4. Phonon Pulser C - AGND
5. Squid C Bias - FEGND
6. QET Bias C - FEGND
7. LED 1 - FEGND
8. FET Heater - FEGND

Test Point 4 (TP-4):

1. Squid D - AGND
2. Phonon D Offset - AGND
3. Qamp Inner Pulser - AGND
4. Phonon Pulser D - AGND
5. Squid D Bias - FEGND
6. QET Bias D - FEGND
7. LED2 - FEGND
8. None - FEGND

- **Number of Charge Channels:** 2

- **Number of Phonon Channels:** 4

- **Interface:**

  STM system interface uses existing WIMP structure controls as it is done in ZIP boards. This structure is as follows:

  - Modules are physically placed into a crate.
  - Each module occupies an unique slot inside the crate. All slots are numbered. The slot number is considered the base address for the module, which is represented by address lines A12…A08 on the J4 backplane connector. This address has to be decoded by all modules in order to determine if data transfer is required to or from a particular module.
  - Address space assigned to a module is divided into 16 sections addressed by four address lines A07…A04.
  - Address space assigned to each section is divided into 16 subsection addressed by four address lines A03…A00. Each subsection actually represents an I/O register or one byte of a memory.
  - Two control signals Read, Write are used to transfer data into and from a CDMS/STM. The Read and Write signals are 50 and 25 μsec long correspondingly. There is no handshake defined in WIMP structure, so address and data are assumed valid on the leading edge of a Write signal and on the trailing edge of a Read signal. Both Read and Write signals are valid when low.

- Data is assumed bi-directional and is transferred back and forth via D15…D0 lines on the J4 backplane connector.

To provide fast data transfers to and from a CDMS/STM, the module employs a standard RS-232 interface, which is carried through a D-Sub connector on module's front panel.

- **Sine Wave Programmable Parameters**

Sine wave is used to check gains in all ZIP amplifiers. Its parameters are as follows:

Type	-	Internally synthesized
Frequency	-	1 KHz
Amplitude	-	from 10 mV to 2.5 V programmable
Resolution	-	8 bits
Output	-	4-pole Low Pass Butterworth Filter with 3 dB point at 5 kHz

- **Event Parameters:**

An event is characterized by issuing four phonon and two charge pulses. The start time of the event coincides with the start time of both charge pulses. All phonon pulses may be delayed from the start by various times. The end time of an event is a sum of the longest delay for one of the phonon signals and its pulse width.

A new event may be issued at any time even when the previous event is in progress. The shortest possible interval between events is 1.5625 μsec.

Each event pulse is characterized by its rise and fall times and by its amplitude. Only the amplitude is programmable for a charge pulse. All phonon pulses have their rise time and amplitude programmable.

Thus, an event has the following characteristics:

- Pulse amplitude - from 0 to 10 V programmable
- Pulse amplitude resolution - 12 bits
- Charge pulse rise time - 1 μsec
- Charge pulse fall time - 40 μsec
- Phonon pulse rise time - from 5 to 50 μsec programmable
- Phonon pulse rise time resolution - 7 bits
- Phonon pulse fall time - 500 μsec
- Phonon pulse delay - from 0 to 100 μsec programmable
- Phonon pulse delay resolution - 10 bits
- Event rate - from 640 KHz to 10 Hz programmable
- Event rate resolution - 12 bits

- **Length of the Event Sequence:** from 1 to 4096 events

- **Front Panel:**

  Front Panel includes:

TP-1, TP-2, TP-3, TP-4	-	Test point BNC connectors
EXT TRIG	-	External trigger BNC connectors
SYNCH OUT	-	Output BNC connector to provide synchronization signal, which mirrors whatever signal is used to run through the Event Sequence
RS-232	-	D-sub RS-232 communication connector
MODE:	-	5-position side actuated DIP switch
Auto		
Run		
Synch		
Single		
Cont		
START	-	Start push button for Single mode and a gate push button for the Cont mode
RESET	-	Reset push button
LED-1, LED-2	-	Indicators to check LED Drivers in ZIP module
MODULE SELECT	-	Indicator showing that module has been addressed
ERROR	-	Module's failure indicator

- **Signal Termination:**

  Signals at TP-1 through TP-4 are terminated as follows:

  Test Point 1 (TP-1):

1. Squid A - 100 Ohms at the input
2. Phonon A Offset - none
3. Qamp Outer - 100 Ohms at the input
4. Phonon Pulser A - none
5. Squid A Bias - 2K Ohms at BNC connector
6. QET Bias A - none
7. Qbias Outer - none
8. Sine Wave - 100 Ohms at BNC connector

  Test Point 2 (TP-2):

1. Squid B - 100 Ohms
2. Phonon B Offset - 100 Ohms

3. Qamp Inner - 100 Ohms
4. Phonon Pulser B - none
5. Squid B Bias - 2K Ohms
6. QET Bias B - 100 Ohms
7. Qbias Outer - 100 Ohms
8. Aux-1 - 100 Ohms

Test Point 3 (TP-3):

1. Squid C - 100 Ohms
2. Phonon C Offset - 100 Ohms
3. Qamp Outer Pulser - none
4. Phonon Pulser C - none
5. Squid C Bias - 2K Ohms
6. QET Bias C - 100 Ohms
7. LED 1 - 2K Ohms
8. FET Heater - none

Test Point 4 (TP-4):

1. Squid D - 100 Ohms
2. Phonon D Offset - 100 Ohms
3. Qamp Inner Pulser - none
4. Phonon Pulser D - none
5. Squid D Bias - 2K Ohms
6. QET Bias D - 100 Ohms
7. LED2 - 2K Ohms
8. Aux2 - 100 Ohms

In addition, the EXT TRIG and SYNCH OUT signals are terminated by 50 Ohms resistors.

- **Power Supply:** - ± 15V (analog), +5V (digital), GNDA (analog ground) , GND (digital ground), FEGND (front end ground)
- **Backplane Connectors:**

The STM employs J1 (AMP 745355-4), J2 (AMP 650948-5), J3 (AMP 745353-4), and J4-1 (AMP650947-5) mating connectors of the ZIP Backplane with the corresponding pin-outs. The pin-outs are shown bellow:

J2

Pin #	Row A	Row B	Row C
1	FEGND	Aux Heater Power +	FEGND
2	FEGND	Aux Heater Power -	FEGND
3	FE +15	FE +15	FE +15
4	FEGND	FEGND	FEGND
5	FE -15	FE -15	FE -15
6	FEGND	FEGND	FEGND
7	FEGND	FEGND	FEGND
8	AGND	AGND	AGND
9	A15+	A15+	A15+
10	AGND	AGND	AGND
11	A15-	A15-	A15-
12	AGND	AGND	AGND
13	10Hz+	AGND	VARIABLE+
14	10Hz-	AGND	VARIABLE-
15	1KHz+	Spare	Trig+
16	1KHz-	Spare	Trig-

J1

Pin #		Pin #		Pin #	
50	Qi Feedback			17	FEGND
		33	FEGND		
49	FET Temp +			16	Qi Drain
		32	FEGND		
48	FEGND			15	Qi Bias
		31	Qi Source		
47	FEGND			14	FEGND
		30	Qo Feedback		
46	FET Heater -			13	FEGND
		29	FEGND		
45	FET Heater +			12	Qo Drain
		28	FEGND		
44	FEGND			11	Qo Bias
		27	Qo Source		
43	FEGND			10	FEGND
		26	LED 2		
42	LED 1			9	FEGND
		25	FEGND		
41	FB Out C			8	SQUID Bias C
		24	FEGND		
40	FEGND			7	QET Bias C
		23	SQUID Bias D		
39	FEGND			6	FEGND
		22	FB Out D		
38	QET Bias D			5	FEGND
		21	FEGND		
37	FB Out A			4	SQUID Bias A
		20	FEGND		
36	FEGND			3	QET Bias A
		19	SQUID Bias B		
35	QET Bias B			2	FB Out B
		18	FEGND		
34	FEGND			1	FEGND

J3

Pin #			Pin #
		Spare 3	13
25	AGND		
		SQUID D Offset Monitor	12
24	AGND		
		SQUID C Offset Monitor	11
23	AGND		
		SQUID B Offset Monitor	10
22	AGND		
		SQUID A Offset Monitor	9
21	AGND		
		Spare 2	8
20	AGND		
		Spare 1	7
19	AGND		
		SQUID D	6
18	AGND		
		SQUID C	5
17	AGND		
		SQUID B	4
16	AGND		
		SQUID A	3
15	AGND		
		Qo Out	2
14	AGND		
		Qi Out	1

J4 (Slots 1 trough 20)

Pin #	Row A	Row B	Row C
1	5 Volts	5 Volts	5 Volts
2	5 Volts	5 Volts	5 Volts
3	5 Volts	5 Volts	5 Volts
4	5 Volts	5 Volts	5 Volts
5	DGND	DGND	DGND
6	D0	DGND	D8
7	D1	DGND	D9
8	D2	DGND	D10
9	D3	DGND	D11
10	D4	DGND	D12
11	D5	DGND	D13
12	D6	DGND	D14
13	D7	DGND	D15
14	DGND	DGND	DGND
15	A0	DGND	A8
16	A1	DGND	A9
17	A2	DGND	A10
18	A3	DGND	A11
19	A4	DGND	A12
20	A5	DGND	A13
21	A6	DGND	A14
22	A7	DGND	A15
23	DGND	DGND	DGND
24	Read	DGND	Write
25	DGND	DGND	DGND
26	5 Volts	5 Volts	5 Volts
27	5 Volts	5 Volts	5 Volts
28	5 Volts	5 Volts	5 Volts
29	Local Address A4	Local Address A3	5 Volts
30	Local Address A2	Local Address A1	Local Address A0
31	DGND	DGND	DGND
32	Bussed Spare 1	Bussed Spare 2	Synch Out

- **Auxiliary Requirements:**

  The STM has to be built as a 9U module modified to CDMS requirements and it has to be a 1 slot wide. Its PCB has to be supported by a stiffener placed alongside and parallel to backplane connectors and by the front panel in order to prevent its bending and warping from being plugged in and out and environmental conditions.

3. Module Structure

The module's block-diagram is shown in Figure 1. It consists of:

- **CPU** – Analog Devices ADuC816 Micro-Converter with Embedded Flash
- **RS-232 Interface** – Analog Devices ADM202 High Speed CMOS RS-232 Drivers/Receivers
- **Power On Reset** – Analog Devices ADM708 μP Supervisory Circuits
- **Interface Controller** – Altera EPM7128 PLD
- **Event Sequence and Configuration Memory** (8K x 16) – IDT IDT7053S/L High Speed Four-Port Static RAM
- **4-pole Low Pass Butterworth Filter** – Linear Technology LTC1063 DC Accurate Clock-Tunable 5th Order Butterworth Low Pass Filter
- **Configuration Controller** – Altera EMP7128 PLD
- **Sequencer** – Altera EPM7128 PLD
- **Signal Router** – REMtech Corp. HGWM Mercury Wetted Latched Relays and Allegro Microsystems, Inc. ULN2803LW 8-bit Darlington Driver
- **Phonon Pulse Generator 1** through **4**
- **Charge Pulse Generator 1** trough **2**

The CPU performs the following operations:

- Provides RS-232 serial interface to the main system for fast download of module configuration and event sequence
- Stores the most current information about module configuration and event sequence into its internal flash memory
- On power on, downloads the contest of its internal flash memory into the Event Sequence and Configuration Memory
- Synthesizes sine wave
- Tests the Event Sequence and Configuration Memory

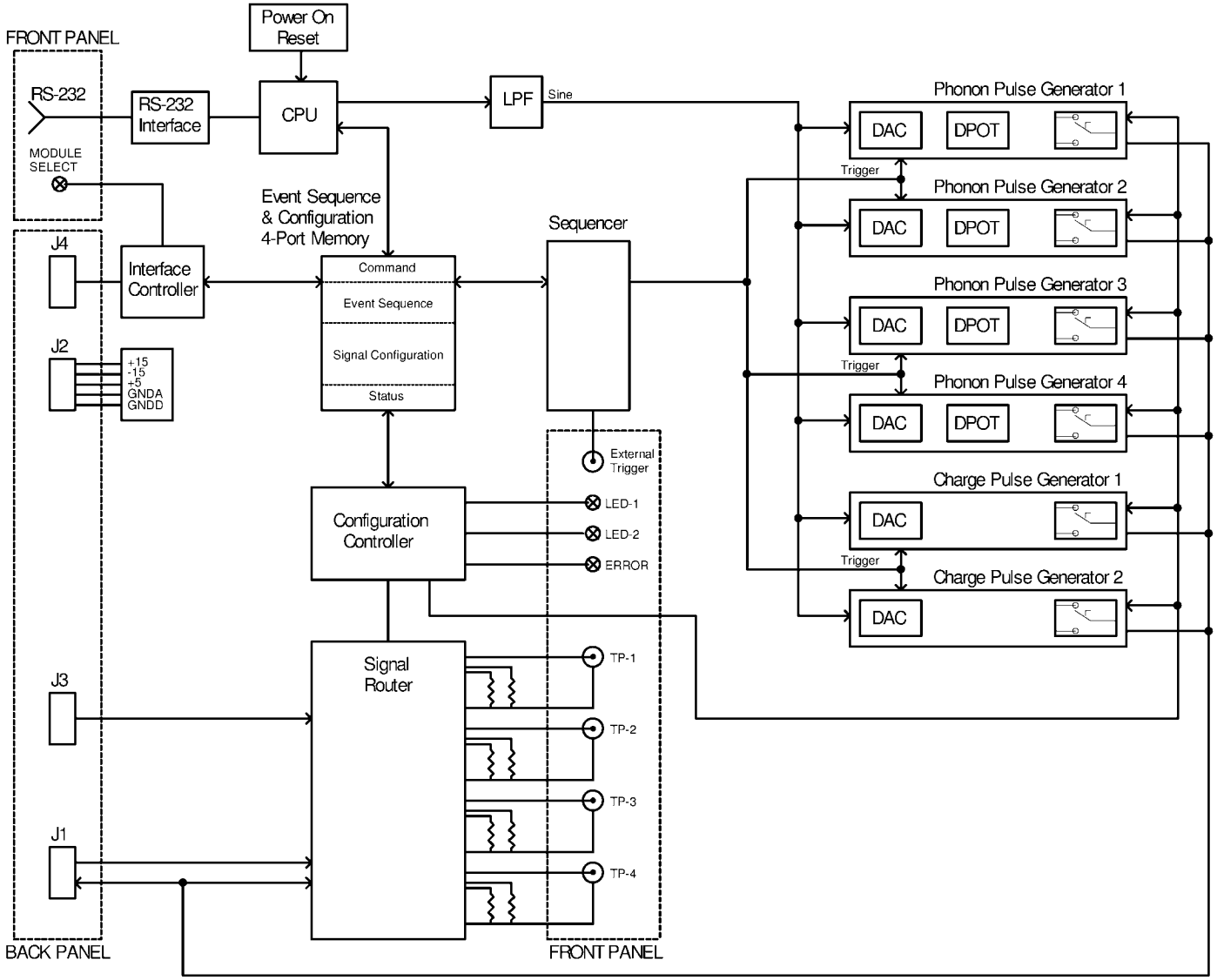

Fig 1. CDMS-STM Block Diagram

The RS-232 Interface converts digital transmit/receive signals from the CPU to standard analog ± 10 V RS-232 signals.

The Power On Reset circuitry produces reset pulse on power on and when power falls lower than 4.5 volts. It also generate the reset pulse from a manual switch on the board.

Interface Controller provides communication between the CDMS STM and the main system, thus allowing to download module configuration and event sequence, to issue commands, and to read module's status. Any information loaded into the module can be read back via module interface.

The Event Sequence and Configuration Memory stores information about module configuration and event sequence. It also holds command and status information.

The 4-pole Low Pass Filter is needed to get rid of higher frequencies in the synthesized sine wave.

The Configuration Controller reads configuration information stored in the memory and configures the Signal Router accordingly. It does this continuously starting on power on and until the module is turned off.

The Signal Router routes tested signals to the test points on the front panel according to configuration information provided by the Configuration Controller. It also provides proper references and terminations for all test points as they are specified above.

The Sequencer sequentially reads event sequence information stored in the memory and relates it to the Phonon and Charge Pulse Generators to produce trigger pulses with given amplitudes, rise times, and delays. A mode, defined either by the main system in command information or by the mode switches on the front panel, controls the behavior of the Sequencer.

The Phonon and Charge Pulse Generators produce actual event (trigger) pulses in accordance with event characteristics given above. All these generators include a DAC to program the amplitude of a pulse and a switch to choose from the sine wave and the event pulse depending on a mode. The switches are controlled by the Configuration Controller. In addition, each of the Phonon Pulse Generators includes a digital potentiometer (DPOT) to control the rise time of the pulse. The proposed schematics for the Phonon and Charge Generators are given in Figures 2, which shows an actual design used to prototype those generators. Shape for a simulated by PSPICE event pulse for the Phonon Pulse Generator is shown in Figure 3 and for the Charge Pulse Generator – in Figure 4.

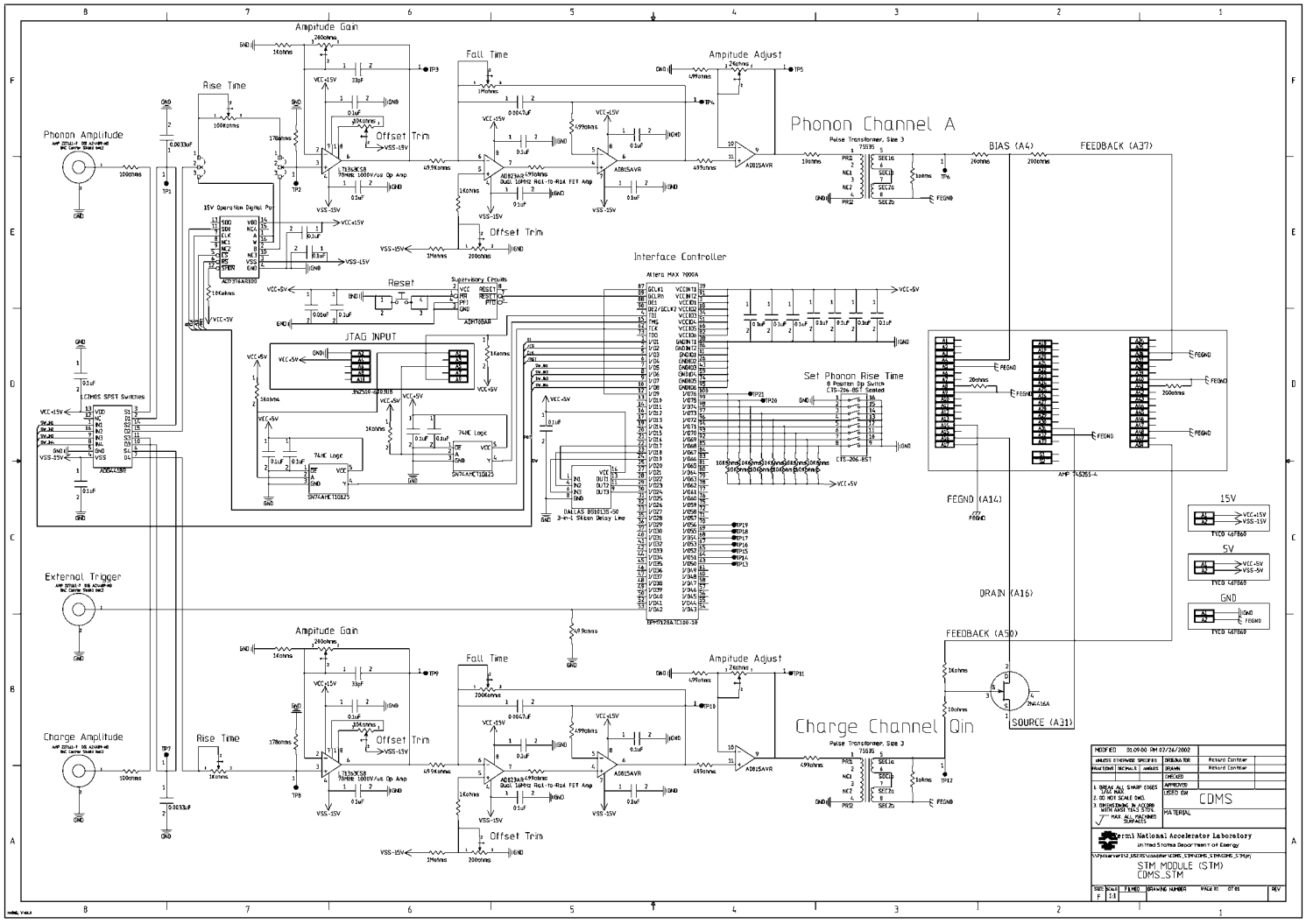

Fig. 2 Phonon and Charge Pulse Generators

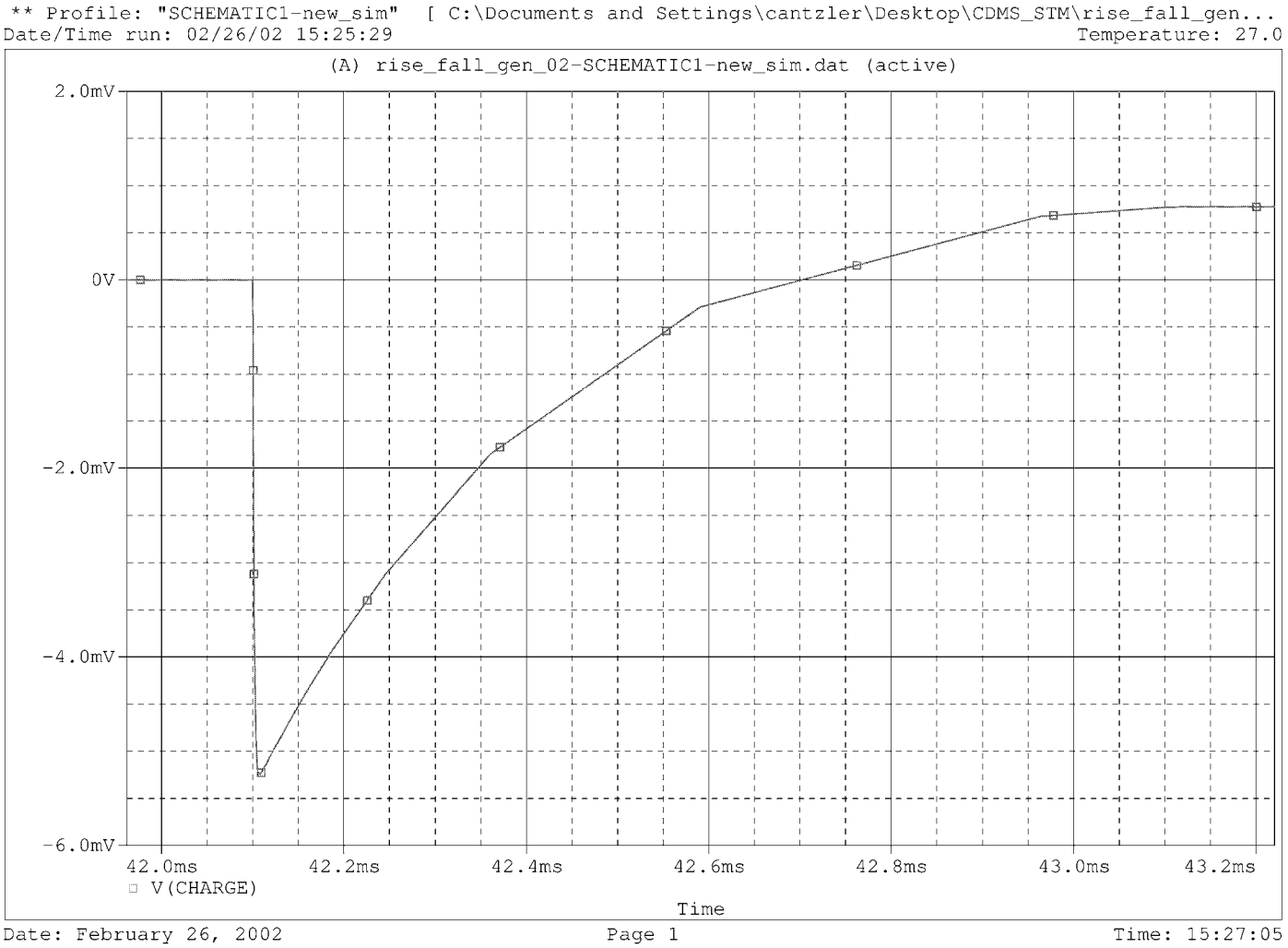

Fig. 3 Phonon Pulse at the Output of the Transformer

Fig. 4 Charge Pulse at the Output of the Transformer

4. Module Description

The module is supposed to work as follows.

On the power on, the CPU performs memory tests and downloads the contents of its internal flash memory into the Event Sequence and Configuration Memory including the contents of the command register. After that, it begins to generate a sine wave used in system debugging to check gains in all amplifiers along the signal path.

The configuration controller reads configuration information and sets relays in the Signal Router and in all Pulse Generators into specified states.

The Sequencer goes through events in the Event Sequence Memory one by one and triggers the Pulse Generators in accordance with timing pattern specified for a particular event. Before triggering the Pulse Generators, the Sequencer downloads amplitude of the event pulse and its rising time into the DAC and DPOT of each generator. Depending on a mode, several different scenarios are possible.

If AUTO mode is chosen, no event pulses go to the back plane to be processed by ZIP modules. Instead, the sine wave is supplied, so gains in all ZIP amplifiers along the signal pass can be checked and adjusted.

In RUN mode, the Sequencer goes trough the entire event sequence defined by a number of events in the Event Sequence Memory. It will do that repeatedly until the STOP bit in the command word is set to be high. If the mode was change during any unfinished event sequence, the Sequencer will first finish this event sequence and only then switch to a new mode.

In SYNCH mode, the Sequencer operates in the same manner as in the RUN mode with the only difference that instead of producing trigger signals for the Pulse Generators by itself, it uses external synchronization pulses to do so. Those pulses can be free running ones because they are gated by START and STOP bits in the command word.

SINGLE mode uses START bit in the command word to trigger the Pulse Generators. By toggling this bit, the external system can make the module to continuously generate the same event.

In CONT mode, the Sequencer works the same way as in SINGLE mode but uses external synchronization pulses instead of the START bit to trigger the generators. Analogous to the SINGLE mode, the produced event is the same for each external pulse. The external pulses can be free running and are gated by START and STOP bits of the command word.

5. Event Sequence Memory Organization

An event is characterized by the following programmable parameters:

Pulse amplitude	-	from 0 to 10 V programmable
Pulse amplitude resolution	-	12 bits
Phonon pulse rise time	-	from 5 to 50 μsec programmable
Phonon pulse rise time resolution	-	7 bits
Phonon pulse delay	-	from 0 to 100 μsec programmable
Phonon pulse delay resolution	-	10 bits
Event rate	-	from 640 KHz to 10 Hz programmable
Event rate resolution	-	12 bits

Four additional pieces of information has to be specified to simplify programming an event sequence. They are

Length of the event sequence	-	12-bit word
Event number	-	12-bit word
Beginning of the event data	-	F555
End of the event data	-	FAAA

The event number from 1 to 4096 is needed to provide a random access to an event in the event sequence. This random access assures an easy change of an event inside the event sequence and its fast reading for verification purposes.

Specifying the length of the event sequence makes it easy to change its length on a fly. If a current event number already exceeds the new sequence length, then the Sequencer will roll over immediately to Event # 1.

Using the beginning and the end of the event data simplifies interface between the module and the main system by using the same address in transferring events from/to the Event Sequence and Signal Configuration Memory. Transmission of both Event Sequence and Signal Configuration takes only one address in the main system address space.

Thus, any event data block transferred into the module's memory has to have the following structure:

F555, L#, E#, Ua, Tr, Td1, Td2, Td3, Td4, Tf, FAAA

where

L#	-	event sequence length (from 1 to 4096)hex
E#	-	event number from (from 1 to 4096)hex;
Ua	-	pulse amplitude calculated as ((A(volts) x 4096) / 10(volts))hex, where A is an actual amplitude;
Tr	-	pulse rising time calculated as (((R(μsec) + 5(μsec)) x 128) / 50(μsec))hex, where R is an actual rise time;
Td1 trough Td4	-	phonon pulse delay calculated as (D(μsec) x 1024 / 100 (μsec))hex, where D is an actual delay;
Tf	-	event rate calculated as ((F(Hz) + 10(Hz)) x 4096 / 640000(Hz))hex

It takes 8 16-bit words to program 1 event. Thus the volume of the Event Sequence Memory is 8 x 4096 = 32768 words = 32K x 16.

Only L# in Event #1 is counted as the event sequence length. L# in all other events are ignored. In order to change the length of the sequence one has to change it in the Event #1. The length of event sequence is always stored in the base address ES-base of the Event Sequence Memory.

When the Interface controller gets the F555 code, it considers incoming information as an event data and starts putting it into the memory. It calculates the beginning address of the event data as ES-base + 1 + (E# - 1) x 8. Thus, Event #1 starts from ES-base + 1, Event #2 – from ES-base + 9, etc.

The Interface Controller stops processing the event when it detects FAAA code. If at that time the Interface Controller finds that the number of processed words for the stored event exceeds 8, then it will issue a communication error signal and will stop accepting event data any further.

6. Signal Configuration Memory

Configuration Memory consists of six 16-bit words, which are configured as follows:

Word 1 – Signals for TP-1 and TP-2
Word 2 – References for TP-1 and TP-2
Word 3 – Terminations for TP-1 and TP-2

Word 4 – Signals for TP-3 and TP-4
Word 5 – References for TP-3 and TP-4
Word 6 – Terminations for TP-3 and TP-4

Considering the base address CM-base for the Configuration Memory, addresses for Word 1 trough 6 will be CM-base + 0 through 5.

Inside each word the data is structured as follows:

Word 1 : Bits 0..7 – Signal for TP-1
Word 1 : Bits 8..15 – Signal for TP-2
Word 2 : Bits 0..7 – Reference for TP-1
Word 2 : Bits 8..15 – Reference for TP-2
Word 3 : Bits 0..7 – Termination for TP-1
Word 3 : Bits 8..15 – Termination for TP-2
Word 4 : Bits 0..7 – Signal for TP-3
Word 4 : Bits 8..15 – Signal for TP-4
Word 5 : Bits 0..7 – Reference for TP-3

Word 5 : Bits 8..15 – Reference for TP-4
Word 6 : Bits 0..7 – Termination for TP-3
Word 6 : Bits 8..15 – Termination for TP-4

Only one bit in each sets of bits assigned to a particular test point can be set high. Same is true for the reference and termination. Thus, only two bits in each word can be set high at a time. Setting a bit high means that the signal assigned to this bit is supposed to be connected to the signal pin of the test point's BNC connector or that a particular ground (digital, analog or the front end ground) is connected to the ground pin of the test point's BNC connector, or that a particular termination resistor is chosen to be connected between signal and ground pins of the test point's BNC connector.

Configuration data transfer into the module's memory has to have the following structure:

5555, Word 1, Word 2, Word 3, Word 4, Word 5, Word 6, 5AAA, where

5555 – Beginning of the configuration data
5AAA – End of the configuration data -

Upon receiving 5555 code, the Interface controller start putting sequential data into corresponding memory addresses. One has to be aware that the order of data in this structure is very important as the Interface Controller does not have capabilities to distinguish between various date and relies on their order to put data into the right location in the Configuration Memory. One has also be aware that the entire structure has to be transferred in order for the module to be configured right. During the transfer, the Interface Controller keeps track of how many words are actually transmitted.

Upon decoding 5AAA code, the Interface Controller checks the number of words received is not equal six. It also checks if the number of bits set high in each word does not exceed two. In any of these case the Interface Controller issues a communication error signal.

7. Command Word Structure

This word specifies a mode of operation and start and end commands. Its structure is as follows:

Bit 0 – AUTO Mode
Bit 1 – RUN Mode
Bit 2 – SYNCH Mode
Bit 3 – SINGLE Mode
Bit 4 – CONT Mode
Bit 5 – Start
Bit 6 – End
Bit 7 through 15 – Undefined

Undefined commands can be specified further in the development.

The command is issued when a corresponding bit is set to high. Only one mode can be specified at a time. If two or more modes are set at the same time, the Interface Controller will issue a mode error signal and will ignore mode setting from the main system choosing instead the mode set by mode switches on the module's front panel.

The command can only be written and it requires a separate address in the main system's address space.

8. Status Information Structure

Status information includes module's mode setting and error information. For the time being only communication, mode, memory, and internal hardware errors are defined. Other status information may be specified further in the development. The structure of the status word is as follows:

Bit 0 – AUTO Mode
Bit 1 – RUN Mode
Bit 2 – SYNCH Mode
Bit 3 – SINGLE Mode
Bit 4 – CONT Mode
Bit 5 – WRITE COMMUNICATION Error
Bit 6 – READ COMMUNICATION Error
Bit 7 – MODE Error
Bit 8 – MEMORY Error
Bit 9 – HARDWARE Error
Bit 10 through 15 – Undefined

Status information can only be read and it requires a separate address in the main system's address space.

9. Module's Memory Organization

Module's full memory is 36K x 16. Its structure is as follows:

Address 0	– Number of Events in the Event Sequence
Address 1 through 32K + 1	– Events from # 1 through 4096
Address 32K + 2 through 7	– Signal Configuration
Address 32K + 8	– Command Register
Address 32K + 9	– Status Register
Address 32K + 10	– Unused

10. Module's System Interface

The main system sees the module as five registers. They are as follows:

Event Sequence and Configuration Write	– write only
Event Sequence and Configuration Read	– read only
Event Number to Read	– write/read
Command	– write only
Status	– read only

Thus the module occupies only five addresses in the main system address space.

Before reading back any event and configuration information, the system has to write an event number into the Event Number to Read register. When afterwards the systems reads the Event Sequence and Configuration register, it will get information in the following order:

1. L#
2. E#
3. Ua
4. Tr
5. Td1
6. Td2
7. Td3
8. Td4
9. Tf
10. Word 1
11. Word 2
12. Word 3
13. Word 4
14. Word 5
15. Word 6

The system must read the Event Sequence and Configuration register 15 times in a row. The Interface Controller will check this and produce a communication error if not all the words were read.

To read another event, the system has to change event number in the Event Number to Read register.

11. RS-232 Protocol

Whatever the main system writes and reads to/from the module via slow parallel interface can be done via much faster RS-232 communication link. The difference is that RS-232 software uses 8-bit wide blocks of information to manipulate instead of 16-bit ones used for

parallel transfers. The actual protocol for data transfer is not presently specified and will be defined further when test software is developed.

12. Comments

Some preliminary work has been done on important parts of the module such as:

- Phonon and Charge Pulse Generators were designed and simulated using Orcad PSPICE environment.
- Prototype of Phonon and Charge Generators is currently under development.
- Sine wave signal generating program for the MCu812 microcontroller was developed and tested.
- 4-pole Low Pass Butterworth Filter for sine signal generation was designed and simulated using Linear Technology FilterCAD environment.